A Wavelet Frames Based Split-Step Method to Model the Tropospheric Long-Range Propagation

Thomas Bonnafont

A Wavelet Frames Based Split-Step Method to Model the Tropospheric Long-Range Propagation

Thomas Bonnafont, *Member, IEEE,*

***Abstract*—Rapid and accurate methods for modeling long-range propagation in the troposphere are of major interest for various applications, including communication, navigation, surveillance, and observation. In this case, the parabolic wave equation model is widely used and commonly solved using split-step methods. In particular, the recent split-step wavelet method has shown promising results in terms of time and memory efficiency. Nonetheless, using a wavelet basis renders the propagation step tricky since the translation invariance property is lost. In this paper, we propose to use wavelet frames in place of the wavelet basis to recover the translation invariance property. This leads to a direct and efficient implementation of the method in the wavelet domain, which can be easily passed on to a GPU. Furthermore, we show that it comes at almost no cost in time or memory efficiency. Numerical experiments are performed to validate the method and highlight its advantages.**

***Index Terms*—Electromagnetic propagation, parabolic wave equation, refraction, split-step method, wavelet, frames**

I. Introduction

FAST numerical methods to model the long-range propagation are necessary to study or optimize the performance of systems, such as radar. In this context, we usually rely on asymptotic methods. Indeed, at the cost of some assumptions, they allow for efficient simulations.

In particular, the parabolic wave equation (PWE) model [1]–[4] is widely used to compute the tropospheric long-range propagation. The latter is a reduced form of the Helmholtz equation where only the forward propagation is accounted for, leading to the possibility of using a wider grid size in the propagation direction [2]. The PWE can be solved using finite difference (FD) schemes [5], [6] or split-step schemes, such as split-step Fourier (SSF) [7]–[9]. Nonetheless, for very large scenarios, SSF is more efficient since the grid size can be larger than the one used in FD schemes [10].

Even if SSF is efficient, it becomes untractable for very large 2D or 3D scenarios. Therefore, wavelet-based split-step (SSW) methods[1] [11]–[16] have been proposed to overcome this problem by using the lower complexity and the sparse representation allowed by the wavelet transform [17]. Indeed, these methods follow the step of SSF, but the free-space propagation is performed in the wavelet domain, leading to lower overall complexity in time and space [14], [15]. Furthermore, a version of SSW fully in the wavelet domain (wSSW) has been proposed [18] to further reduce its computation time.

T. Bonnafont is with the Lab-STICC, UMR CNRS 6285, ENSTA, Institut Polytechnique de Paris, Brest, France e-mail: thomas.bonnafont@ensta.fr.

Manuscript received XX.XX.XXXX

[1]In particular, in [11], [12], Douvenot *et al.* proposed an open source version of split-step wavelet.

For free-space propagation, wSSW [18] is thus very efficient in computation time. Nonetheless, all the operators need to be adapted to the wavelet basis, where the translation invariance property is not preserved. This leads to going back in the spatial domain for the ground, increasing the size of the wavelet scattering operator, and increasing the error due to the wavelet phase screen operator. In addition, at each propagation step, each coefficient level needs to be treated differently, which makes this method unattractive for GPU implementation.

In image processing, to avoid the loss of the translation property, wavelet frames are widely used [17], [19], [20]. Furthermore, in electromagnetic (EM), Gabor frames are also utilized to model the propagation [21], [22], leading to efficient representation of the EM field. In this work, to overcome some limitations of SSW and wSSW, we introduce split-step framelet (SSfW), a split-step method based on the wavelet frames representation.

Here, the main idea of wSSW is preserved. Indeed, we replace the wavelet basis by frames and calculate the associated scattering operator. Furthermore, using the Haar family, we derive a direct wavelet local image method, i.e., the wavelet counterpart of [14, Section 4.4]. Since frames preserve the translation property, we can directly introduce the terrain model in the wavelet domain. In addition, the refraction is also accounted for in the wavelet domain, as in [18], but without any loss. This leads to a split-step method fully in the wavelet domain, with a complexity of the same order as SSW or wSSW. The latter can also be easily implemented on GPU. Finally, as for SSW [23], we can derive an *a priori* error bound.

The remainder of the paper is organized as follows. Section II presents the split-step framelet method. Section III is devoted to the numerical simulations. They aim to validate the method, and we also propose a case study. Finally, Section IV concludes the paper and gives perspectives for future works.

II. The split-step framelet method

A. A quick reminder on the parabolic wave equation

Throughout this paper, we assume an $\exp(j\omega t)$, with ω the angular frequency, time variation, and that the refractive index, denoted by n, is slowly varying with the propagation direction. In addition, we work in the Cartesian coordinate system (x, z) with x the propagation axis and z the altitude. Furthermore, the considered domain is $[0, x_{\max}] \times [0, z_{\max}]$ since the field at $x = 0$ is assumed to be known and only the above-ground propagation is of interest. Thus, the considered source is located at $x_s \leq 0$ and $z_s \geq 0$.

In the context of long-range propagation, one widely used model is the wide-angle parabolic wave equation (PWE) [2].

The latter is a reduced form of the Helmholtz equation, where only the forward propagation is accounted for, and corresponds to

$$\frac{\partial u}{\partial x} = -jk_0\left(\sqrt{\frac{\partial^2}{\partial z^2}+k_0^2}-k_0\right)u - jk_0(n-1)u, \quad (1)$$

with k_0 the free-space wave number, and u the reduced field [2], [8]. It shall be noted that equation (1) is only true in a paraxial cone of $45°$ around the propagation direction x since the wide-angle approximation has been chosen. It shall be noted that in general with the PWE we use the modified refractivity $M = 10^6(n-1)+10^6\frac{z}{R_e}$, in M-units, that accounts for the earth flattening transformation. Nonetheless, here we choose to keep the notation n throughout the paper for readability.

For obvious numerical reasons, we need to discretize the domain to solve equation (1). This discretization is performed with steps Δx and Δz leading to $N_x = x_{\max}/\Delta x$ and $N_z = z_{\max}/\Delta z$ points over each axis. Thus we denote by $u[p_x, p_z]$ the discrete reduced field at (p_x, p_z) and by $u_x[\cdot]$ it semi-discretized version over the z-axis.

B. Brief introduction on wavelet frames

Since they are at the core of the proposed method, this section is devoted to a rapid introduction to wavelet frames and their useful properties.

Frames are a generalization of a basis where the elements are not necessarily linearly independent, i.e., we have a redundant representation. In more details, a countable family $(\chi_p)_p$ is said to be a frame if

$$A\|u\|^2 \leq \sum_p \langle u, \chi_p\rangle \leq B\|u\|^2, \quad (2)$$

with $0 < A \leq B$ and $\langle\cdot,\cdot\rangle$ the associated scalar product. This implies a normalization of a factor 2^L in the transform. Here , we only use energy-conservative tight-frame, leading to $A = B = 1$. In addition, we only consider wavelets for the family χ_p. To obtain the wavelet coefficient in a frame, we can use the stationary wavelet transform (SWT), which has a complexity greater than the fast wavelet transform (FWT), i.e., $O(N\log(N))$ instead of $O(N)$. Nonetheless, the SWT can easily be implemented on GPU.

Indeed, as for SSW, the idea is to use the highly sparse representation allowed by the wavelet representation to accelerate the method. For a complete introduction to wavelets, we refer to [17], and for their use in computational electromagnetics to [24]. In a few words, a wavelet family is constructed by dilating, over L levels, and translating a mother wavelet, i.e., a zero-mean function. These dilations allow for covering the spectrum until the lowest frequency. To account for this last part, a scaling function, or father wavelet, which is not of zero-mean, is added to the family.

Here, one can thus wonder: *why not use wavelet basis as in [14], [15], [18]*? The main reason is that, contrary to wavelet bases, with wavelet frames, the translation invariance property is kept [17]. This has made them quite attractive in image theory, for example. Here, this property seems also of interest, since contrary to [18] it allows us to construct directly the wavelet counterpart of all the operators, as will be seen later on. Furthermore, it seems quite normal to consider a transformation where the coefficients associated with a translated version of the field are not different from the non-translated one.

Finally, the wavelet frame transform over L levels of a field u will be denoted from hereafter as U. Thus, we have $U[l, p]$ the coefficients at level $l \in [0, L]$, with 0 corresponding to the scaling function, and $p \in [0, N_z - 1]$ the position.

C. Split-step framelet (SSfW)

This part is devoted to the introduction of the split-step framelet (SSfW) method. Since Zhou *et al.* [14] have shown that the wavelet family has almost no impact on the accuracy, we choose here the Haar family. Indeed, the latter allows for efficient theoretical calculations since, to the author's knowledge, that is the only discrete wavelet family for which wavelet and scaling functions are antisymmetric and symmetric. Nonetheless, most of the results can be generalized to other families.

1) Overview of the method: As for SSW [14], [15], SSfW is an iterative method that computes the field marching in on distances over the x-axis. The main difference is that we do not go back and forth between the wavelet and spatial domain, but stay in the wavelet domain, as wSSW [18], to expedite the calculations.

Therefore, the method works as follows. First, we compute the wavelet coefficients associated with u_0 as

$$U_0 = \mathbf{C}_{V_s}\mathbf{S}_w u_0, \quad (3)$$

where $\mathbf{S}_w$ is the stationary wavelet transform (SWT), and $\mathbf{C}_{V_s}$ a hard threshold compression, with threshold V_s. Then, to compute $U_{x+\Delta x}$ from U_x, the following steps are performed:

1) To account for the ground, the wavelet local image method, described in Sec. II-C3, is applied.
2) The coefficients are propagated considering a layer of free space
$$U^{\text{fs}}_{x+\Delta x} = \mathbf{P}_{V_p}U_x, \quad (4)$$
where $\mathbf{P}_{V_p}$ is the space wavelet-to-wavelet scattering operator. This step is described in more detail in Sec. II-C2.
3) The effects of the refraction and terrain are accounted for through a phase-screen, $\mathbf{L}$, and relief, $\mathbf{R}$, operators as
$$U_{x+\Delta x} = \mathbf{R}\mathbf{L}U^{\text{fs}}_{x+\Delta x}. \quad (5)$$
Both are defined in Sec. II-C4.
4) Compression with hard threshold V_s is applied to only keep the important information and reduce the computation time.

All these steps are performed until we reach $x_{\max}$ at which point an inverse stationary wavelet transform is applied to recover $u_{x_{\max}}$. Therefore, even if the SWT has a larger complexity than the FWT, it is only used 2 times here.

2) Free space propagation in the wavelet domain: This paragraph presents the frame counterpart of the wavelet basis scattering operator introduced in [15]. Indeed, the main idea is the same, we need to know how the wavelet propagates in free space from x to $x + \Delta x$.

First, as proposed in [15], we construct a library of local operators. To do so, we use SSF to propagate one wavelet, denoted by χ_l^0, of each level $l \in [0, L]$ from x to $x + \Delta x$ leading to $\chi_l^{\Delta x}$ of size N_{supp}. Then, we apply an SWT and a compression with a hard threshold V_p to obtain the associated propagated wavelet coefficients which leads to

$$\mathbf{P}_{V_p}[l] = \mathbf{C}_{V_p}\mathbf{S}_w\chi_l^{\Delta x}. \tag{6}$$

The main difference with [15] is that no spatial translations of the wavelet need to be considered since in a wavelet frame we are translation invariant. This leads to a reduction of the size of the library to only L elements of size $L \times N_{\text{supp}}$, with N_{supp} the support size of the wavelet after propagation [15]. This library corresponds to the wavelet scattering operator.

Second, this operator is used to propagate each non-zero coefficient of U_x from x to $x + \Delta x$. This step amounts to

$$U_{x+\Delta x}^{fs}[l'] = \sum_l U_x[l] \otimes \mathbf{P}_{V_p}[l, l'], \tag{7}$$

where $\otimes$ corresponds to the convolution. Note that a direct convolution can be performed here, with the translation invariance, whereas in SSW [15] translations are needed for certain levels, i.e., all $l > 1$. This is an important difference, since it allows for an efficient GPU implementation.

3) The wavelet local image method: For SSW, Zhou *et al.* [14] have introduced the local image method to account for ground with a limited amount of points. Here, we propose a version of this method directly in the wavelet domain. In a few words, using the filter bank of the Haar wavelets and their symmetry and anti-symmetry properties, we can directly calculate the image field wavelet coefficient from U_x. In Fig. 1, we provide an illustration for the proposed wavelet local image method.

Let $N_{\text{im}} \ll N_z$ be the number of points in the image layer, and denote by U_x^{im} the image field wavelet coefficients. The latter is a matrix of size $L \times (N_z + N_{\text{im}})$. Let us also denote by Γ the Fresnel reflection coefficient. First, for all levels the coefficient above N_{im} are

$$\forall l \in [0, L], \forall p_z \in [0, N_z]\, U_x^{\text{im}}[l, N_{\text{im}} + p_z] = U_x[l, p_z], \tag{8}$$

which correspond to the above-ground coefficients that are thus unchanged. Second, for the maximum level $l = L$, the below the ground coefficients correspond to

$$\forall p_z \in [0, N_{\text{im}} - 1],\, U_x^{\text{im}}[l, p_z] = -\Gamma U_x[l, N_{\text{im}} - p_z - 1].$$

Indeed, for the Haar system for level L the filter is antisymmetric and given by $h_L = [-1/2, 1/2]$. Then for any level $0 < l < L$, first we compute the number of coefficients that can not be directly calculated from the above the ground coefficients

$$N_0 = (2^{L-l} - 1) \times 2,$$

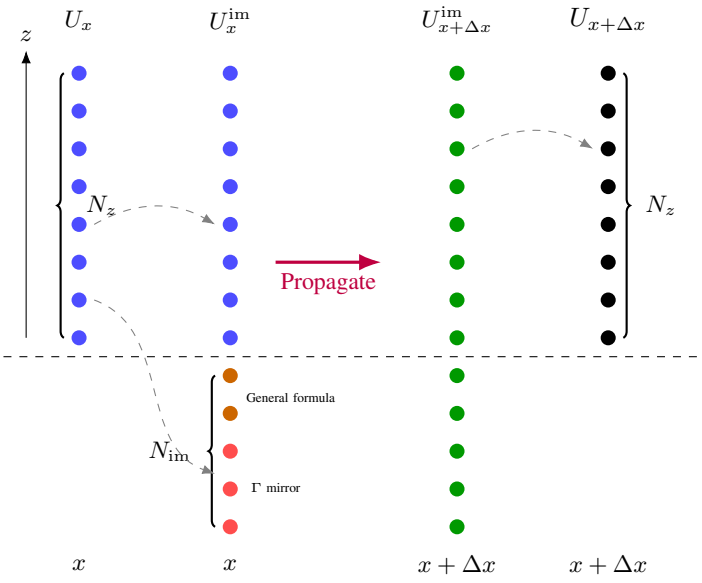

Fig. 1. Schematic representation of the wavelet local image method.

which correspond to the number of 0 in the associated filter. Second, $\forall p_z \in [0, N_{\text{im}} - 1 - N_0]$ the coefficients are computed as

$$U_x^{\text{im}}[l, p_z] = -\Gamma U_x[l, N_{\text{im}} - p_z - 1 - N_0].$$

For the other coefficients, the following general formula can be used $\forall k \in [1, N_0]$

$$U_x^{\text{im}}[l, N_{\text{im}} - k] = -\frac{\Gamma}{2^{L-l-1}} \left(\sum_{i=N_0/2+1-k}^{N_0-k-1} \sum_{j=0}^{i-1} U_x[L, j] - \sum_{i=1}^{N_0/2-k} \sum_{j=0}^{i-1} U_x[L, j] \right) + \sum_{i=1}^{k} \sum_{j=0}^{i-1} U_x[L, j].$$

Finally for $l = 0$, we get $N_0 = (2^L - 1) \times 2$. Furthermore, since the associated filter is symmetric we get $\forall p_z \in [0, N_{\text{im}} - 1 - N_0]$ the coefficients are computed as

$$U_x^{\text{im}}[l, p_z] = \Gamma U_x[l, N_{\text{im}} - p_z - 1 - N_0],$$

and the other coefficients are determined by $\forall k \in [1, N_0]$

$$U_x^{\text{im}}[l, N_{\text{im}} - k] = -\frac{\Gamma}{2^{L-1}} \sum_{i=1}^{N_0-k-1} \sum_{j=0}^{i-1} U_x[L, j] - \sum_{i=1}^{k} \sum_{j=0}^{i-1} U_x[L, j].$$

In Appendix A we give a sketch of the proof and example of the formula for $L = 1$, $L = 2$ and $L = 3$ for the interested readers. Furthermore to avoid discontinuity at the upper boundary, the coefficient near N_z, i.e., $\lfloor N_0/2 \rfloor$ coefficients, shall be modified accordingly. The proposed wavelet local image method can only be used for wavelet families for which the filters are symmetric and/or anti-symmetric, thus only for the Haar system. Indeed, otherwise, no direct formula can be calculated, and a linear system must be solved, which would

reduce the computation efficiency. Thus, for other families, we recommend using the strategy of Zhou *et al.* [18] and come back in the spatial domain on $N_{\rm im}$ points to account for the ground, even if it reduces the computational time efficiency. In addition, it can only be used for frames since the transformation over a wavelet basis introduces decimation of some coefficients at each level.

4) Accounting for the refraction and the terrain: Using wavelet frames, accounting for the refraction is easy. Indeed, contrary to [18] due to the redundancy of $(\chi_p)_p$ and the translation invariance, we can directly use the phase screen operator [2], [8] defined as

$$\mathbf{L}[p_z, p_z] = \exp\left(-jk_0(n_x - 1)\right). \tag{9}$$

The latter is a diagonal matrix of size $N_z \times N_z$ that is then multiplied by each level vector of coefficients to account for the refraction at each step. One advantage is that frames allow us to account for the refraction in the wavelet domain without loss, whereas in [18] it introduces a decrease in accuracy.

Finally, to account for the terrain, we use a staircase model [2] that represents the relief as a series of ascending and descending rectangles. Therefore, this amounts to translations in the spatial domain, and since frames are translation invariant, we can directly derive this model in the wavelet domain. Note that to avoid any discontinuity near the edges due to the 0-padding of translations, we wrap $\lfloor N_0/2 \rfloor$ lower coefficients at the top.

5) Error bound and complexity:

a) Error bound: We can use the results from [23] for the error bound. Indeed, since wavelets are used, the same theoretical calculations can be performed. Since tight frames with $A = B = 1$ are used, we have a normalization factor of $1/\sqrt{2}^{L-1}$ for the transform. Therefore, it modified the threshold introduced in [23] as follows

$$V_s = \frac{\delta_{N_x}^{\max}}{2N_x\sqrt{2}^{L-1}}\|u_0\|_\infty, \text{ and } V_p = \frac{\delta_{N_x}^{\max}}{2N_x\sqrt{2}^{L-1}}\|P\|_{\max},$$

where $\delta_{N_x}^{\max}$ is a chosen maximum level of accuracy at the last iteration.

Now, let us derive the space, i.e., the amount of necessary memory space with the input size, and time complexity , i.e., amounts of computation time with the input size, of the proposed method to compare it to SSW [14], [15] and wSSW [18]. Hereafter, we denote by N_w and N_{V_p} the number of non-zero coefficients after compression for the field and the scattering operator in SSfW, while they are denoted by N_w^* and $N_{V_p}^*$ for SSW and wSSW. In general, we have $N_w \geq N_w$ and $N_{V_p} \geq N_{V_p}^*$, but high compression rates lead to $N_w \sim N_w$ and $N_{V_p} \sim N_{V_p}^*$.

b) Comparison with SSW: First let us compare to SSW [15]. For SSfW, we apply the stationary wavelet transform only once, leading to a complexity of $\mathcal{O}(N_z \log(N_z))$. At the same time, with SSW, we go back and forth in the spatial and wavelet domains at each step, leading to a complexity of $\mathcal{O}(N_x(N_z + N_{\rm im}))$. Considering the scattering operator, for SSfW this step amounts to a time complexity of $\mathcal{O}(LN_{\rm supp}\log(N_{\rm supp}))$ while its spatial complexity is $\mathcal{O}(LN_{V_p})$ where N_{V_p} is the number of coefficients after compression. For SSW, the time complexity is of the same order, as the same strategy is used, which requires a propagation step of SSF, the limiting part here. Nonetheless, in terms of spatial complexity, the SSW operator requires $\mathcal{O}(LN_{V_p}^* N_t^l)$, since translated versions of the field are needed due to the wavelet basis dilations at each level [15]. For the propagation step, the time complexity is of order $\mathcal{O}(N_w N_{V_p})$ for SSfW and $\mathcal{O}(N_w^* N_{V_p}^*)$ for SSW. Thus, they are of the same order; nonetheless, in terms of implementation, SSW requires translating and choosing the local operator [15] before performing the convolution, which can lead to a slight advantage for SSfW. Nonetheless, regarding spatial complexity, since the method operates fully in the wavelet domain, only the wavelet coefficients are saved, resulting in a spatial complexity of $\mathcal{O}(N_w)$ for SSfW. In contrast, by returning to the spatial domain at each step, the spatial complexity for propagation with SSW is $\mathcal{O}(N_z)$.

c) Comparison with wSSW: Second, we compare to wSSW [18]. In this case, the FWT is only applied once, leading to a time complexity of $\mathcal{O}(N_z)$ for the transformation step, thus wSSW has a slight advantage for the latter compared to SSfW, but it becomes negligible for large propagation domains. Considering the scattering operator, it is constructed as a matrix [14], [18] without using the advantage of the small support size of the wavelet, the time complexity associated is thus $\mathcal{O}(LN_z\log(N_z))$ with $N_z \gg N_{\rm supp}$. In addition, the propagators are saved in a sparse matrix of size $N_z \times N_z$ with $N_z N_{V_p}^*$-terms, leading to a higher spatial complexity of $\mathcal{O}(LN_z N_{V_p}^*)$ compared to SSfW. Nonetheless, the propagation step amounts to a multiplication of a sparse vector with a sparse matrix, which is thus $\mathcal{O}(N_w^* N_{V_p}^*)$, but in general more computationally efficient than the local convolution performed in SSW. Finally, to apply the local image method, with wSSW, one shall go back in the spatial domain over $2N_{\rm im}$ points at each step, leading to an overall computational cost of $\mathcal{O}(N_x N_{\rm im})$.

d) Summary on the spatial and time complexity: To conclude, we propose to sum up the comparison between SSfW, qnd SSW and wSSW in Table I. From this table we can conclude that all methods have a similar complexity, except that SSfW allows is slightly better in terms of memory usage. In addition, we can clearly see that the computation time will increase with L increasing. Furthermore, another advantage of SSfW is that all the operations are directly implementable on GPU, which can lead to a better advantage in terms of computation time. Indeed, all the proposed operations have GPU-efficient counterparts, while for SSW or wSSW it would be more costly due to the translation needed for the propagation step or to the high dimensionality of the scattering operator, respectively.

III. Numerical experiments

This section is devoted to numerical simulations. First, the wavelet local image method is validated and tested for different levels L. Second, we show that the method works well in a simple case with a propagation over a PEC ground by

TABLE I
COMPARISON OF THE TIME AND SPACE COMPLEXITY OF SSFW, SSW, AND WSSW FOR ALL THE IMPORTANT STEPS. THE FIRST FOUR CORRESPOND TO THE TIME COMPLEXITY, WHILE THE LAST TWO ARE THE SPACE COMPLEXITY ASSOCIATED WITH SAVING THE FIELD AND THE SCATTERING OPERATOR.

Method	**SSfW**	SSW	wSSW
Transform	$\mathcal{O}(N_z log(N_z))$	$\mathcal{O}(N_x(N_z+N_{im}))$	$\mathcal{O}(N_z)$
init P	$\mathcal{O}(LN_{\text{supp}}\log(N_{\text{supp}}))$	$\mathcal{O}(LN_{\text{supp}}\log(N_{\text{supp}}))$	$\mathcal{O}(LN_z\log(N_z))$
Propa	$\mathcal{O}(N_xN_wN_{V_p})$	$\mathcal{O}(N_xN_w^*N_{V_p}^*)$	$\mathcal{O}(N_xN_w^*N_{V_p}^*)$
Ground	-	-	$\mathcal{O}(N_xN_{im})$
u	N_w	N_z	N_w^*
P	$\mathcal{O}(LN_{V_p})$	$\mathcal{O}(LN_t^lN_{V_p}^*)$	$\mathcal{O}(LN_zN_{V_p}^*)$

comparing it to SSF. We also compare SSfW to SSW in terms of computation time for different L. Third, we propose a more realistic test case, where a dielectric ground, an atmospheric duct, and relief are accounted for. Finally, we propose to use the method for the refractivity from clutter (RFC) problem. In addition, all the codes used for this part are available at the following link: https://gitlab.ensta-bretagne.fr/bonnafth/ssfw, and SSfW computations are performed on CPU without parallelization for a fair comparison. As stated in Section II-C3, the wavelet local image method only works for the Haar family; thus, it is the only one used. In addition, numerous studies on the family have been performed in [14], [25] and have shown that it has almost no impact on the accuracy.

A. Validation of the wavelet local image method

The main goal of this section is to validate the proposed wavelet local image method and to compare its accuracy for different levels $L \in [1, 3]$.

To do so, we construct a field, which is a complex source point (CSP) [26]. The latter is placed at $x_s = -50$ m and $z_s = 10$ m and has a width of $W_0 = 5$ m. The domain is of size 256 m in altitude with a step size of $\Delta z = 0.5$ m. We consider a PEC for the ground, and a image layer of size $0.25N_z$. In all cases, a compression of 10^{-5} is performed in the wavelet domain. For the constructed CSP field, we compute its image using the image theorem to compare to the field retrieved after applying a SWT to the CSP field, the wavelet local image method and an inverse SWT. We plot the obtained fields in Fig. 2.

Furthermore, we compute the maximum of difference between the fields constructed using the wavelet local image method and the true image field. We obtain -95 dB, -91.5 dB and -81.5 dB for $L = 1$, $L = 2$ and $L = 3$, respectively. As can be seen, this errors occurs near the edges and is believed to be due to the compression and the fact that for a higher level the span of the wavelets increase. Nonetheless, the proposed method is validated and we thus encourage to keep $L \leq 2$ to avoid propagation of numerical artifacts over large domains. Note that in [18], they gave the same guideline for the level.

B. Validation of the method

In this first test, we compute the propagation from an antenna at $f_0 = 300$ MHz. The latter is modeled as a complex source point (CSP) [26] placed at $x_s = -50$ m and $z_s = 20$ m with a width of $W_0 = 5$ m. The considered domain is $[0, 10000] \times [0, 512]$ m^2. We set $\Delta x = 50\lambda = 50$ m

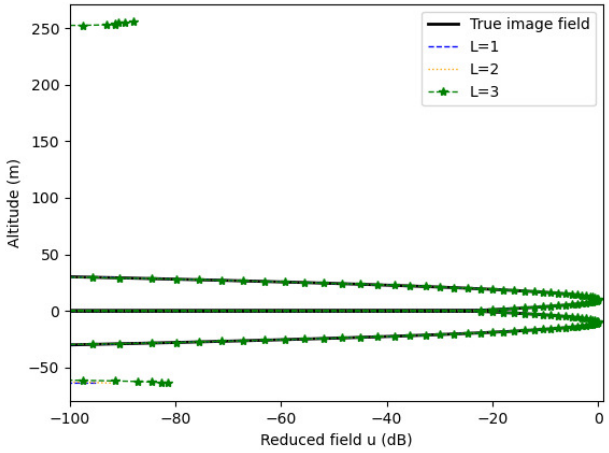

Fig. 2. Reduced field u in dB with its image field computed either with the image theorem (black plain line) or with the wavelet local image method (dotted lines) for different levels.

and $\Delta z = \lambda/2 = 0.5$ m for the discretizations. Finally, we consider a PEC ground.

The propagation is computed with SSfW and SSW for different levels, either $L = 1$ or $L = 2$, and the accuracy with respect to SSF and the computation time of both methods are compared. These results are reported in Table II. Furthermore, for $L = 2$ we plot the field on the last iteration computed with SSfW, SSW, and SSF in Fig. 3. The one obtained with the latter method is in solid line, while the two others correspond to the dashed stars and the dotted plus lines for SSfW and SSW, respectively. The same legend is used for their respective errors with SSF.

TABLE II
COMPARISON OF SSW AND SSFW REGARDING COMPUTATION TIME AND ACCURACY.

	SSW		**SSfW**	
maximum level (L)	1	2	1	2
Computation time (s)	0.27	0.62	0.26	0.54
Maximum error (dB)	-52.15	-52.48	-69.07	-57.98

First, SSfW performs well in this case, yielding lower errors than SSW when using the compression formula from Section II. As expected, its computation time is comparable to that of SSW for $L = 1$, and becomes lower as L increases.

In this case, the method is thus validated. However, we can see that $L = 1$ yields a smaller error than $L = 2$ for SSfW and similar accuracy to SSW, but with reduced computation time. Indeed, from the previous section, the error for $L = 2$ is

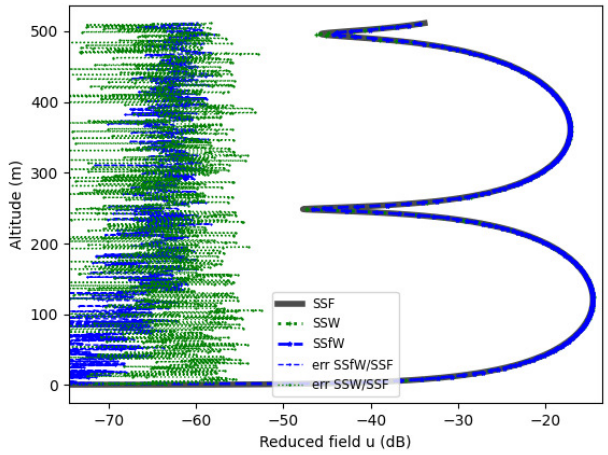

Fig. 3. Reduced field u in dB computed at the $x = x_{\max}$ with SSF, SSW, and SSfW and the associated difference in dB between SSW and SSF, and SSfW and SSF.

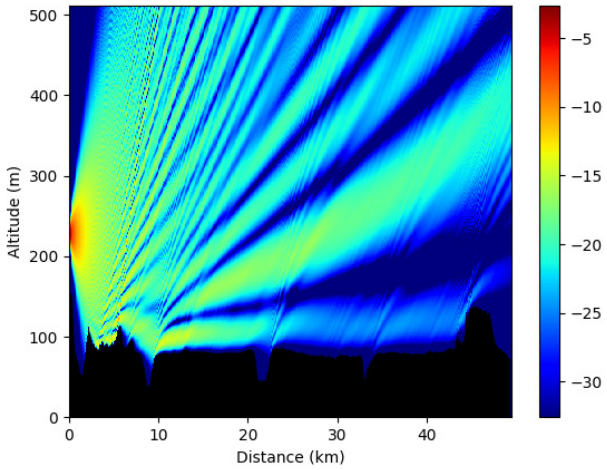

Fig. 4. Reduced field u in dB computed with SSfW on the overall domain.

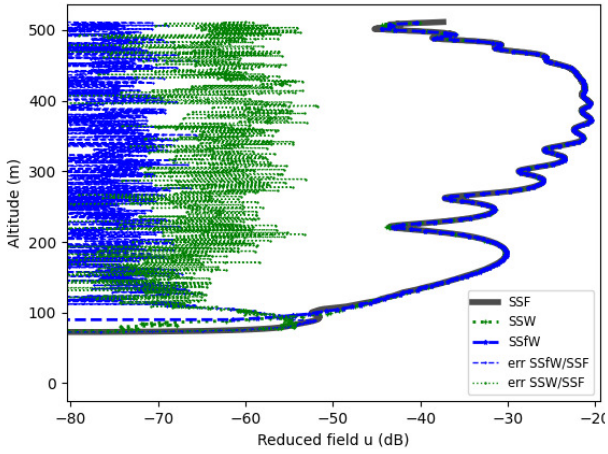

Fig. 5. Reduced field u in dB computed at the $x = x_{\max}$ with SSF, SSW, and SSfW and the associated difference in dB between SSW and SSF, and SSfW and SSF.

mainly due to the artifact near the edges, thus we recommend to keep $L = 1$ in general, and do it from now. For $L = 2$, in this case we obtain a lower error with SSfW but it is not statistically significant and in different cases we could obtain closer difference with SSF at the last iteration.

C. A realistic test case

Now, we consider a more realistic test case where terrain, a dielectric ground, and refraction are considered. The source's parameters remain the same except for its altitude, which is now $z_s = 70$ m above the ground. The goal is to model the propagation from Palaiseau (ENSTA) to Fontainebleau (Castle), two French cities distant by 49.5 km, leading to $x_{\max} = 49.5$ km. Furthermore, we consider $z_{\max} = 512$ m. We consider the following discretization: $\Delta = 100\ \lambda = 100$ m and $\Delta z = \lambda/2 = 0.5$ m. The terrain between the two cities is accounted for using the data provided by IGN [27]. We also consider a dielectric ground with $\varepsilon_r = 20$ and $\sigma = 0.02$ S/m. Finally, a tropospheric duct modeled by a tri-linear refractive index [28] is accounted for. For the latter, we use the following parameters: $M_0 = 330$ M-units, $z_b = 20$ m, $z_t = 50$ m, and $c_0 = 0.118$ M-units/m and $c_2 = -0.5$ M-units/m.

In Fig. 4, we plot the reduced field in dB computed with SSfW on the overall domain. Then, in Fig. 5, we propose, as before, to show the fields computed at $x = x_{\max}$ with SSF, SSfW, and SSW to compare them with SSF as the reference. In this case, the computation time is 0.23 s for SSfW and 0.25 s for SSW. Furthermore, the maximum difference with SSF at $x_{\max}$ is -52.01 dB for SSfW and -51.75 dB for SSW. Here, we are in a scenario close to the test case of [18], with $N_x \sim 500$ and $N_z = 1024$, and we obtain a similar computation time as expected from the complexity study. In addition, in [18], they compared a matrix-based version of SSW [14], and wSSW showed a similar speed-up as the one of the local SSW [15], [29] that is used here, denoted as SSW, which further highlights the results of the complexity study. In addition, Zhou *et al.* [18] reported a high increase in error, almost 10 dB when a tropospheric duct is considered, which is not the case here.

First, in Fig. 4, we can see the effects of the terrain with the interference pattern and of the refraction with the bending of the rays. In addition, we can see that the results with SSfW are comparable to SSW in terms of accuracy, while being slightly faster. Furthermore, contrary to [18], accounting for the refraction does not add any error. In addition, relief and ground conditions are directly accounted for in the wavelet domain here. This shows that wavelet frames allow for an easy and efficient implementation in a split-step method. Furthermore, a GPU extension of SSfW is direct.

D. On the use of SSfW for the refractivity from clutter problem

Finally, since SSfW is fast, we use it in a simple case of refractivity from clutter (RFC) problem, see [30]–[32] for more details about RFC. The goal is to retrieve the refractive index n from measured field data on a vertical at $z_{\max}$. Here, we show that on noisy artificial data, computed with SSF[2], and using blackbox optimization [33], [34], we can retrieve a good approximation of n using SSfW. Indeed, the time efficiency

[2]Indeed, the artificial data are computed on a smaller grid and with another numerical method to avoid the so-called "inverse crime".

of the latter allows us to use it as a direct solver in blackbox optimization, since in many EM practical problems we do not have access to the gradient. Note that blackbox optimization methods allow tuning fewer hyperparameters than genetic algorithms, as used in [32], to converge.

The goal is to retrieve a tri-linear refractive profile n with the parameters of the previous test. The measurements correspond to artificial data computed with SSF on the vertical at $x_{\max}$ with a fine grid over z (a step size of $\lambda/4$). Furthermore, they are performed at two locations $z_s = 30$ m and $z_s = 90$ m. Then, these measures are corrupted with a Gaussian noise (with different levels). This leads to the following cost function to minimize

$$J(n) = \|u^{\text{SSfW}}_{z_s=30} - u^{\text{mes}}_{z_s=30}\|_2 + \|u^{\text{SSfW}}_{z_s=90} - u^{\text{mes}}_{z_s=90}\|_2 + \alpha\|n\|_2,$$

where $u^{\text{mes}}_{z_s=.}$ is the vector of noisy measures and $u^{\text{SSfW}}_{z_s=.}$ the field computed with SSfW for a refractive index n and an transmitter location of z_s, and the last term is a Tikhonov regularization. Here, since this part is not the core of the paper, we set $\alpha = 10^{-4}$. The following parameters are used to initialize the refractive index n_0: $M_0 = 320$ M-units, $z_b = 10$ m, $z_t = 35$ m, and $c_0 = 0.3$ M-units/m and $c_2 = -0.3$ M-units/m. To conclude the scenario, the other parameters remain the same as the first test, except that $z_{\max} = 256$ m. In Fig. 6, we plot the real M-index we want to retrieve and the ones that have been computed from the blackbox optimization inversion for different levels of noise.

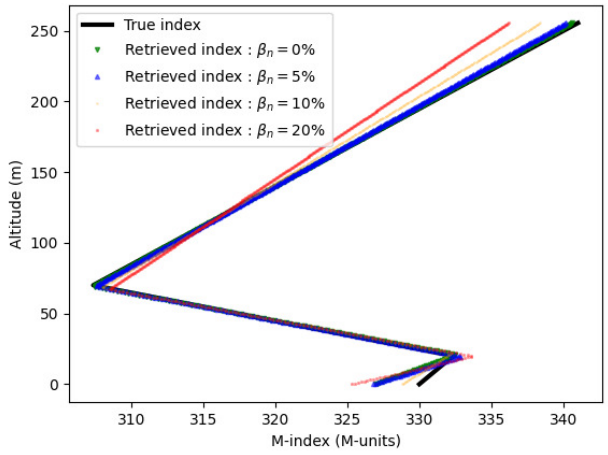

Fig. 6. Retrieved M-index with the SSfW-based black box inversion for different levels of noise: 0%, 5%, 10%, and 20%.

First, for all these tests, the inversion took at most 60 s, showing the efficiency of the proposed approach[3]. This is mostly because the proposed SSfW method is fast. Furthermore, we can see that the inversion is accurate even with noise. Obviously, for $\beta_n = 0\%$, with β_n the noise level, we retrieve almost exactly M except for a slight difference in the boundary area $z \leq z_b$. This difference is true for all levels of noise and due to the fact that the transmitting antennas are all placed such that $z_s \geq z_b$. In addition, even for $\beta_n = 10\%$, the results are quite accurate, showing the robustness of the approach. Nonetheless, for a higher noise level, only a part of the M-index is well retrieved, and robust methods, e.g., based on a Karhunen-Loeve decomposition, shall be considered in this case. Furthermore, more transmitting antennas would result in a higher accuracy, and since the loss function consists of a sum of errors for all measurements, it could be parallelized for efficiency.

This test underlines that SSfW can be used in a realistic case, here the RFC problem, where an optimization step is performed. This opens the way to its use in other similar scenarios, e.g., antenna location and parameters optimization for communication applications.

[3]It shall be noted that the blackbox optimization has been used as is without parallelization and on a normal desktop computer.

IV. Conclusion

In this article, we proposed a fully wavelet frames-based split-step method to model the long-range propagation, while accounting for the terrain, the ground composition, and the refraction.

First, the SSfW method has been introduced. The latter is based on the use of wavelet frames in place of a wavelet basis to keep the translation invariance property, which seems natural in our case. Indeed, this leads to more coefficients to account for, but since high compression is possible in the wavelet domain, we show that the complexity is not increased compared to SSW. In addition, we provide a wavelet local image method that allows for directly accounting for a dielectric ground in the wavelet domain. Due to the translation invariance property, the phase screen and the terrain operator are seamlessly generalized in the wavelet-frame domain. Since all the operators are performed in the wavelet domain, the method can be efficiently implemented on GPU. Finally, numerical tests in the VHF band have been proposed to first validate the method; second, to show its advantages in a realistic scenario; and third, to apply it in an RFC scenario together with blackbox optimization using its time efficiency.

Nonetheless, this work opens up several interesting perspectives. First, we are currently working on an efficient GPU implementation of the method to decrease its computation time further. Second, following the approach of SSW [16], we're exploring a generalization to the 3D case. A fully GPU-based wavelet-frames split-step method could help address current performance limitations. Finally, we are currently working on using the method in realistic scenarios, such as a robust version of the proposed method for the RFC problem, or using it to optimize communication links in a maritime environment.

Appendix

A. On the wavelet local image method

The proof of the general formulas used in the proposed wavelet local image method, see Section II-C3, derives directly from calculating and comparing the wavelet coefficients of $u_x = [0, u_1, u_2, \cdots, u_{N_z-1}]$ and $u_x^{\text{im}} = [\Gamma u_{N_{\text{im}}-1}, \cdots \Gamma u_1, 0, u_1, u_2, \cdots, u_{N_x-1}]$, where u_x^{im} is the local image field. Indeed, the wavelet coefficients are obtained by convolution with the following filters for the Haar system

$$h_l = [1/2, 0, \cdots, 0, -1/2], \text{ and } g_l = [1/2, 0, \cdots, 0, 1/2].$$

The number of coefficients we need to recalculate at each level depends on the number of zeros in the filters, i.e., $N_0 = 2 \times (2^{L-l} - 1)$. In addition, the sign difference $\pm$ between all the levels and $l = 0$ comes from the fact that the wavelets are anti-symmetric while the scaling function is symmetric.

As an example, we give here the formula for $L \in [1, 2]$. For all cases, the above-ground coefficients are unchanged as stated in the wavelet local image method. Then, for a maximum level $L = 1$, the general formula leads to

$$\forall p_z \in [0, N_{\rm im} - 1],\, U_x^{\rm im}[L, p_z] = -\Gamma U_x[L, N_{\rm im} - p_z - 1],$$
$$\forall p_z \in [0, N_{\rm im} - 1],\, U_x^{\rm im}[0, p_z] = \Gamma U_x[0, N_{\rm im} - p_z - 1].$$

If $L = 2$ then the rest of the coefficients are calculated as follows by simplifying the general expressions. For $l = L$, we get

$$\forall p_z \in [0, N_{\rm im} - 1],\, U_x^{\rm im}[l, p_z] = -\Gamma U_x[l, N_{\rm im} - p_z - 1].$$

For $l = 1$, the image coefficients are

$$U_x^{\rm im}[l, p_z] = -\Gamma U_x[l, N_{\rm im} - p_z - 3],\, p_z \in [0, N_{\rm im} - 3],$$
$$U_x^{\rm im}[l, N_{\rm im} - 2] = -\Gamma \frac{\sum_{i=0}^{1} U_x[L, i]}{4} - (\Gamma - 1)\frac{U_x[L, 0]}{2},$$
$$U_x^{\rm im}[l, N_{\rm im} - 1] = \frac{\sum_{i=0}^{2} U_x[L, i]}{4} - (\Gamma - 1)\frac{U_x[L, 0]}{2}.$$

For $l = 0$, we obtain

$$U_x^{\rm im}[l, p_z] = \Gamma U_x[l, N_{\rm im} - p_z - 3],\, p_z \in [0, N_{\rm im} - 3],$$
$$U_x^{\rm im}[l, N_{\rm im} - 2] = -\Gamma \frac{\sum_{i=0}^{1} U_x[L, i]}{4} - (\Gamma + 1)\frac{U_x[L, 0]}{2},$$
$$U_x^{\rm im}[l, N_{\rm im} - 1] = -\frac{\sum_{i=0}^{2} U_x[L, i]}{4} - (\Gamma + 1)\frac{U_x[L, 0]}{2}.$$

REFERENCES

[1] M. A. Leontovich and V. A. Fock, "Solution of the problem of propagation of electromagnetic waves along the earth's surface by the method of parabolic equation," *J. Phys. USSR*, vol. 10, no. 1, pp. 13–23, 1946.

[2] M. Levy, *Parabolic equation methods for electromagnetic wave propagation*. Edison, NJ, USA: IET, 2000.

[3] G. D. Dockery, "Modeling electromagnetic wave propagation in the troposphere using the parabolic equation," *IEEE Transactions on Antennas and Propagation*, vol. 36, no. 10, pp. 1464–1470, 2002.

[4] G. Apaydin and L. Sevgi, *Radio wave propagation and parabolic equation modeling*. Hoboken, NJ, USA: John Wiley & Sons, 2017.

[5] M. D. Collins, "A split-step Padé solution for the parabolic equation method," *The Journal of the Acoustical Society of America*, vol. 93, no. 4, pp. 1736–1742, 1993.

[6] R. Brent and J. Ormsby, "Scalar electromagnetic propagation modelling using parabolic equations and the split-step Pade approximation," *Journal of Physics A: Mathematical and General*, vol. 28, no. 7, p. 2065, 1995.

[7] D. J. Thomson and N. Chapman, "A wide-angle split-step algorithm for the parabolic equation," *The Journal of the Acoustical Society of America*, vol. 74, no. 6, pp. 1848–1854, 1983.

[8] D. Dockery and J. R. Kuttler, "An improved impedance-boundary algorithm for fourier split-step solutions of the parabolic wave equation," *IEEE Transactions on Antennas and Propagation*, vol. 44, no. 12, pp. 1592–1599, 1996.

[9] J. R. Kuttler and R. Janaswamy, "Improved Fourier transform methods for solving the parabolic wave equation," *Radio Science*, vol. 37, no. 2, pp. 1–11, 2002.

[10] M. D. Collins, R. J. Cederberg, D. B. King, and S. A. Chin-Bing, "Comparison of algorithms for solving parabolic wave equations," *The Journal of the Acoustical Society of America*, vol. 100, no. 1, pp. 178–182, 1996.

[11] R. Douvenot, "SSW-2D: some open-source propagation software introducing split-step wavelet and wavelet-to-wavelet propagation techniques," in *2023 IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communications (APWC)*. IEEE, 2023, pp. 020–024.

[12] R. Douvenot, "SSW-2D: Split-step wavelet in 2D." [Online]. Available: https://github.com/remi-douvenot/ssw-2d

[13] A. Iqbal and V. Jeoti, "An improved split-step wavelet transform method for anomalous radio wave propagation modeling," *Radioengineering*, vol. 23, no. 4, pp. 987–996, 2014.

[14] H. Zhou, R. Douvenot, and A. Chabory, "Modeling the long-range wave propagation by a split-step wavelet method," *Journal of Computational Physics*, vol. 402, p. 109042, 2020.

[15] T. Bonnafont, R. Douvenot, and A. Chabory, "A local split-step wavelet method for the long range propagation simulation in 2D," *Radio Science*, vol. 56, no. 2, pp. 1–11, 2021.

[16] ——, "Split-step wavelet with local operators for the 3D long-range propagation," in *2021 15th European Conference on Antennas and Propagation (EUCAP)*. IEEE, 2021, pp. 1–5.

[17] S. Mallat, *A wavelet tour of signal processing*. Amsterdam, Netherlands: Elsevier, 1999.

[18] H. Zhou, A. Chabory, and R. Douvenot, "A fast wavelet-to-wavelet propagation method for the simulation of long-range propagation in low troposphere," *IEEE Transactions on Antennas and Propagation*, vol. 70, no. 3, pp. 2137–2148, 2021.

[19] B. Han, "On dual wavelet tight frames," *Applied and Computational Harmonic Analysis*, vol. 4, no. 4, pp. 380–413, 1997.

[20] C. K. Chui, W. He, and J. Stöckler, "Compactly supported tight and sibling frames with maximum vanishing moments," *Applied and computational harmonic analysis*, vol. 13, no. 3, pp. 224–262, 2002.

[21] J. J. Maciel and L. B. Felsen, "Gabor-based narrow-waisted Gaussian beam algorithm for transmission of aperture-excited 3d vector fields through arbitrarily shaped 3D dielectric layers," *Radio Science*, vol. 37, no. 2, pp. 1–9, 2002.

[22] D. Lugara, C. Letrou, A. Shlivinski, E. Heyman, and A. Boag, "Frame-based Gaussian beam summation method: theory and applications," *Radio Science*, vol. 38, no. 2, pp. 27–1, 2003.

[23] T. Bonnafont, R. Douvenot, and A. Chabory, "Determination of the thresholds in the split-step wavelet method to assess accuracy for long-range propagation," *URSI Radio Science Letters*, vol. 3, 2021.

[24] T. K. Sarkar, M. Salazar-Palma, and M. C. Wicks, *Wavelet applications in engineering electromagnetics*. Norwood, Massachusetts, USA: Artech House, 2002.

[25] T. Bonnafont, "Modeling the atmospheric long-range electromagnetic waves propagation in 3D using the wavelet transform," Ph.D. dissertation, Université Paul Sabatier-Toulouse III, 2020.

[26] G. A. Deschamps, "Gaussian beam as a bundle of complex rays," *Electronics letters*, vol. 7, no. 23, pp. 684–685, 1971.

[27] "Elevation lines data of the "Institut nationale de l'informations Géographique et Forestière" (IGN)," https://www.geoportail.gouv.fr/, accessed: 01-04-2025.

[28] E. E. Gossard and R. G. Strauch, *Radar observation of clear air and clouds*. Amsterdam, Netherlands: Elsevier, 1983.

[29] T. Bonnafont, R. Douvenot, and A. Chabory, "A speed up of split-step wavelet for the computation of long range propagation," in *2020 14th European Conference on Antennas and Propagation (EuCAP)*. IEEE, 2020, pp. 1–5.

[30] P. Gerstoft, L. T. Rogers, J. L. Krolik, and W. S. Hodgkiss, "Inversion for refractivity parameters from radar sea clutter," *Radio science*, vol. 38, no. 3, 2003.

[31] A. Karimian, C. Yardim, P. Gerstoft, W. S. Hodgkiss, and A. E. Barrios, "Refractivity estimation from sea clutter: An invited review," *Radio Science*, vol. 46, no. 06, pp. 1–16, 2011.

[32] J. Heo, D. Jang, H. Choo, D.-Y. Na, and Y. B. Park, "Real-time duct prediction and propagation analysis using genetic algorithm-based rfc and parabolic equation method," *IEEE Access*, 2024.

[33] L. Acerbi and W. J. Ma, "Practical Bayesian optimization for model fitting with Bayesian adaptive direct search," *Advances in Neural Information Processing Systems*, vol. 30, pp. 1834–1844, 2017.

[34] G. S. Singh and L. Acerbi, "PyBADS: Fast and robust black-box optimization in Python," *Journal of Open Source Software*, vol. 9, no. 94, p. 5694, 2024. [Online]. Available: https://doi.org/10.21105/joss.05694